ИЗВЕСТИЯ ВЫСШИХ УЧЕБНЫХ ЗАВЕДЕНИЙ
№ 1 ФИЗИКА 2013
УДК 519.712

А.Ю. МАТРОСОВА, М.Л.ГРОМОВ, В.В.ЖАМНОВ, Е.А.НИКОЛАЕВА

ПОИСК ПРОСТОЙ ЦЕПИ ОГРАНИЧЕННОЙ ДЛИНЫ В УСЛОВИЯХ ДИНАМИЧЕСКИ ИЗМЕНЯЮЩИХСЯ ВЕСОВ ДУГ[1]

Для взвешенного орграфа предлагается алгоритм поиска простой орцепи, длина которой не превосходит заданного значения. Веса дуг орграфа могут динамически изменяться. Предполагается, что доля таких дуг незначительна. Алгоритм является модификацией алгоритма Дейкстры. Обсуждается возможность применения предлагаемого алгоритма при передаче информации в коммутационных сетях.

Ключевые слова: *ориентированный граф, вес дуги, длина простой цепи.*

Рассматривается ориентированный граф (орграф) G, его дугам приписаны целые неотрицательные числа, называемые весами. Заданы две вершины α, β графа. Ставится задача найти простую орцепь μ[1], длины не более h, соединяющую вершину α с вершиной β, в условиях динамического изменения весов дуг орграфа G. При неизменности весов дуг орграфа эта задача решается известным алгоритмом Дейкстры [2,3]. В данной работе предлагается модификация этого алгоритма, основанная на предположении, что веса дуг графа остаются неизменными в процессе выполнения k шагов алгоритма(назовем эти вычисления k-шаговой процедурой). Это значит, что каждая k-шаговая процедура работает с теми данными, которые имелись непосредственно перед ее реализацией. Более того, мы полагаем, что изменяются веса небольшого по мощности подмножества дуг.

1. Модификация шага алгоритма Дейкстры

Сначала опишем предлагаемую модификацию шага алгоритма Дейкстры. Простой орцепью будем называть ормаршрут(путь), в котором вершины не повторяются. Обозначим через $d(x_i)$ расстояние от вершины α до вершины x_i. Под расстоянием здесь и далее будем понимать сумму весов дуг простой орцепи, соединяющей α с x_i, то есть длину простой орцепи. Простых орцепей может быть несколько. Под минимальным расстоянием будем понимать минимальную сумму весов дуг простой орцепи, соединяющей α с x_i. Пусть $a(x_j, x_i)$ вес дуги из x_j в x_i. Веса дуг представляются неотрицательными числами. Если дуга отсутствует или направлена из x_i в x_j, то ее вес полагаем равным ∞. В качестве вершины x_j при реализации шага алгоритма Дейкстры используется отмеченная на предыдущем шаге алгоритма вершина, минимальное расстояние до которой от начальной вершины α уже вычислено. Предлагаемая модификация шага алгоритма заключается в следующем.

Находим все смежные с x_j вершины (x_j – отмеченная на рассматриваемом шаге вершина) и пересчитываем расстояния до них по формуле Дейкстры:

$$\underbrace{d(x_i)}_{\substack{\text{новое}\\ \text{растояние}}} = \min \quad \text{старое растояние},\ d(x_j) + a(x_j, x_i)$$

Если новое расстояние оказалось меньше старого, то отмечаем дугу (x_j, x_i). В алгоритме Дейкстры отмечалась только одна дута.

Эта дуга обеспечивала глобальное в пределах шага минимальное значение среди всех неотмеченных, в том числе смежных с x_j, вершин. Отмеченная дуга заходит в вершину x^*, которая объявляется отмеченной для реализации следующего шага алгоритма Дейкстры: $x^* = x_j$ на следующем шаге.

Сведем предлагаемую модификацию алгоритма к последовательному выполнению k-шаговых процедур, т.е. после выполнения k шагов возвращаемся назад, если в соответствующем фрагменте графа изменились веса дуг.

2. Алгоритм поиска простой орцепи ограниченной длины

[1] *Работа проводилась при финансовой поддержке Министерства образования и науки Российской федерации в рамках соглашения №14.В37.21.0622 от 16.08.2012г.*

1. Выполняем k модифицированных шагов очередной k-шаговой процедуры. Если время поиска исчерпано, или найдена отмеченная вершина, до которой минимальная длина орцепи от вершины α больше h, переходим к п. 6. (На каждом шаге k-шаговой процедуры, как и в алгоритме Дейкстры, находится единственная отмеченная вершина). Если в процессе реализации k-шаговой процедуры $x^* \neq \beta$, переходим к п. 2 алгоритма. Если $x^* = \beta$, переходим в п. 4.

2. Анализируем измененные веса дуг сети. Если среди дуг с измененными весами нет отмеченных дуг, возвращаемся в п. 1 и выполняем следующую k-шаговую процедуру, взяв в качестве начальной отмеченной вершины ту, которая была вычислена в конце предыдущей k-шаговой процедуры. Иначе переходим в п. 3

3. Для вершин, в которые заходят отмеченные дуги с измененными весами, находим отмеченную вершину x^{**}, которая соответствует шагу с меньшим номером. Предполагается сплошная нумерация шагов алгоритма в пределах всех выполненных k-шаговых процедур. Возвращаемся в п. 1, выбрав в качестве начальной отмеченнойвершины k-шаговой процедуры вершину x^{**}.

4. Простая орцепь, проходящая по отмеченным ребрам через отмеченные вершины, является искомой простой орцепью μ. Анализируем изменение весов дуг орграфа G после нахождения этой орцепи. Если измененные веса отсутствуют среди ее дуг, или длина орцепи не превосходит h, то переходим к п. 5. Иначе возвращаемся к п. 3 алгоритма, пользуясь полученной информацией об измененных весах дуг. При этом рассматриваются все дуги, отмеченные в процессе вычисления простой орцепи μ.

5. Решение найдено.

6. Не существует простой орцепи длины не более h.

Приведем пример. Пусть в графе выделены входной и выходной полюсы, то есть рассматривается сеть, являющаяся 1,1-полюсником. Требуется найти простую орцепь длины не более h, $h = 15$, соединяющую полюсы сети. На рис. 1 представлена сеть с взвешенными ребрами. Жирными отрезками отмечена простая орцепь длины L между полюсами сети, которая находится алгоритмом Дейкстры (не модифицированным), если бы веса дуг не изменялись, $L = h$.

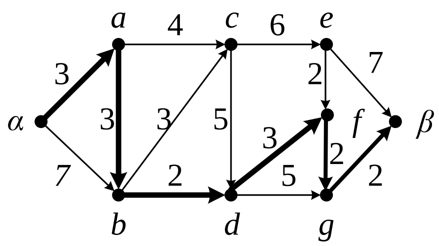

Рис. 1. Простая орцепь μ в условиях постоянных весов ребер сети

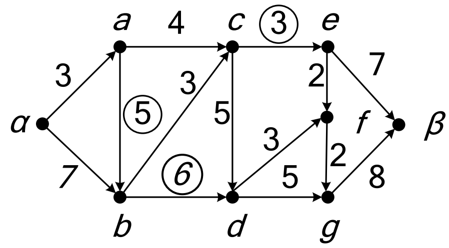

Рис. 2 Состояние сети после выполнения первой k-шаговой процедуры

Число шагов k-шаговой процедуры равно двум, $k = 2$. Воспользуемся рис. 1 для выполнения k-шаговой процедуры.

1. $d(\alpha) = 0$, $y = \alpha$, $d(a) = \min(\infty, 0+3) = 3$, $d(b) = \min(\infty, 0+7) = 7$, $d(c) = \ldots = d(\beta) = \infty$,
отмеченная вершина: a, $y = a$, отмеченные дуги: (α, a), (α, b).

2. $d(y) = 3$, $y = a$, $d(c) = \min(\infty, 3+4) = 7$, $d(b) = (7, 3+3) = 6$, $d(d) = \ldots = d(\beta) = \infty$,
отмеченная вершина: b, $y = b$, отмеченные дуги: (a, c), (a, b).

Выполнив 2 шага, переходим к анализу сети с измененными весами (рис. 2). Измененные веса дуг отмечены в кружках. Поскольку отмеченная дуга (a, b) изменилась, возвращаемся ко 2-му шагу алгоритма, используя сеть на рис 2.

3. $d(y) = 3$, $y = a$, $d(c) = \min(\infty, 3+4) = 7$, $d(b) = (7, 3+5) = 7$, $d(d) = \ldots = d(\beta) = \infty$,
отмеченная вершина: b, $y = b$, отмеченные дуги: (a, c), (α, b).

4. $d(y) = 7$, $y = b$, $d(c) = \min(7, 7+3) = 7$, $d(d) = (\infty, 7+6) = 13$, $d(e) = \ldots = d(\beta) = \infty$,
отмеченная вершина: c, $y = c$, отмеченные дуги: (b, d) (a, c). Анализируем сеть с измененными весами дуг (рис. 3).

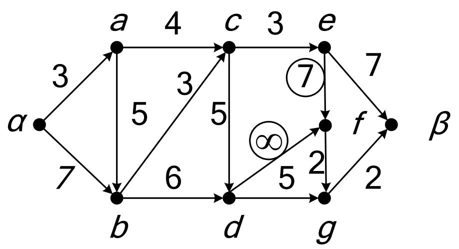

Рис. 3. Состояние сети после выполнения второй *k*-шаговой процедуры

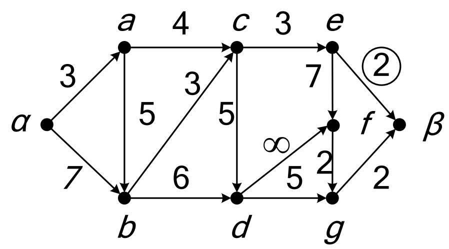

Рис. 4. Состояние сети после выполнения третьей *k*-шаговой процедуры

Изменения в сети не затронули отмеченных на предыдущих шагах дуг. Продолжаем вычисления из вершины, достигнутой на шаге 4.

5. $d(y) = 7$, $y = c$, $d(e) = \min(\infty, 7+3) = 10$, $d(d) = (13, 7+5) = 12$, $d(f) = \ldots = d(\beta) = \infty$, отмеченная вершина: *e*, $y = e$, отмеченные дуги: (c, e), (c, d).

6. $d(y) = 10$, $y = e$, $d(f) = \min(\infty, 10+7) = 17$, $d(\beta) = (\infty, 10+7) = 17$, $17 > h$, $d(g) = \infty$, отмеченная вершина: *f*, $y = f$, отмеченные дуги: (e, f), (e, β).

Анализируем сеть с измененными весами дуг (рис. 4). Изменения в сети затронули отмеченную дугу (e, β). Возвращаемся в вершину *e*. Продолжаем вычисления, используя рис. 4.

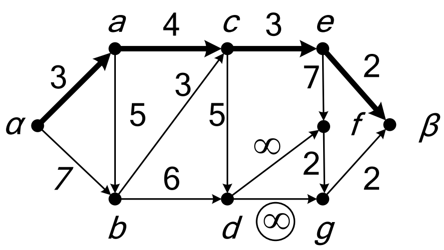

Рис5. Простая орцепь μ в условиях переменных весов ребер сети

7. $d(y) = 10$, $y = e$, $d(f) = \min(\infty, 10+7) = 17$, $17 > h$, $d(\beta) = (\infty, 10+2) = 12$, β – выходной полюс сети, $12 < h$. Решение найдено, простая орцепь представляется последовательностью вершин: α, *a*, *c*, *e*, β.

Анализируем сеть в момент нахождения простой цепи (рис. 5). Изменения в сети не затронули найденной простой цепи, которая на рис. 5 выделена жирными отрезками.

Предложенная модификация алгоритма Дейкстры может найти применение при решении следующей задачи.

Рассматривается телекоммуникационная сеть *S* с одним входным полюсом (CDN) и одним выходным полюсом (пользователь). Внутренние вершины сети (маршрутизаторы) характеризуются различными параметрами, которые учитываются интегрированной характеристикой веса дуги графа (неотрицательным числом). Требуется найти путь, длины не более *h*, соединяющий входной полюс сети с выходным полюсом, то есть найти путь, по которому абоненту можно доставить контент за приемлемое время и с приемлемым качеством. Усложняющим условием решения этой задачи является тот факт, что веса дуг графа могут меняться во время проведения вычислений.

СПИСОК ЛИТЕРАТУРЫ

1. Р. Уилсон. Введение в теорию графов // М:Мир – 1977. – 208с.
2. Н. Кристофидес. Теория графов. Алгоритмический подход // М:Мир – 1978. – 429с.
3. Н.И. Костюкова. Графы и их применение. Комбинаторные алгоритмы для прогрммистов.М.-2007.-311с.

Матросова Анжела Юрьевна, д.ф.-м.н., профессор;
Громов Максим , к.ф.-м.н. доцент;
Жамнов Вадим Владимирович, ст. преподаватель;
Николаева Екатерина Александровна, к.т.н., доцент

A. MATROSOVA, M.GROMOV,V. ZHAMNOV, E.NIKOLAEVA

FINDING A PATH WITH RESTRICTED LENGTH ON THE ASSAMPTION OF CHANGING GRAPH AGE WEIGHTS

The modification of Dijikstra algorithm is suggested on the assumption of changing graph edge weights that takes place during determing path length. It is noticed that developed algorithm may be employed for fast and high quality transmission of a content from CDN to user.

Keywords: *directed graph, weight of edge, path length.*